Amendments to the Claims

1. (Currently amended) A method of performing a recoverable operation on a message queue in response to a request by a caller in an information handling system, said method comprising the steps of:

storing a use count for said message queue indicating a count of tasks accessing said message queue;

storing a use count flag for said caller indicating whether said caller has acquired a lock on said message queue;

comparing said use count with a previously read use count; and

if said use count matches said previously read use count, then, atomically with said comparing step, updating said use ~~count;~~ count to indicate a new count of tasks accessing said message queue and ~~atomically with updating said use count,~~ updating said use count flag to indicate whether said caller has acquired a lock on said message queue.

2. (Previously presented) The method of claim 1 in which said recoverable operation is a locking operation, said step of updating said use count comprising the step of incrementing said use count, said step of updating said use count flag comprising the step of updating said use count flag to indicate that said caller has acquired a lock on said message queue.

3. (Previously presented) The method of claim 1 in which said recoverable operation is an unlocking operation, said step of updating said use count comprising the step of decrementing said use count, said step of updating said use count flag comprising the step of updating said use count flag to indicate that said caller has released a lock on said message queue.

4. (Cancelled)

5. (Currently amended) The method of ~~claim 4~~ claim 1 in which said use count is stored in a message queue table having an entry for said message queue.

6. (Previously presented) The method of claim 5 in which said message queue table also stores a pointer to said message queue, said method comprising the further step of:

comparing said pointer with a previously read pointer atomically with said updating steps, said updating steps being performed only if said pointer matches said previously read pointer.

7. (Previously presented) The method of claim 5 in which said message queue table also stores an identifier of said message queue.

8. (Original) The method of claim 1 in which said use count flag is stored in a control block for said caller.

9. (Previously presented) The method of claim 8 in which said control block for said caller also contains an identifier of said message queue.

10. (Original) The method of claim 1 in which said updating steps are performed by executing a single atomic instruction that updates said use count and, concurrently therewith, updates said use count flag.

11-23. (Cancelled)

24. (Currently amended) Apparatus for performing a recoverable operation on a message queue in response to a request by a caller in an information handling system, comprising:

means for storing a use count for said message queue indicating a count of tasks accessing said message queue;

means for storing a use count flag for said caller indicating whether said caller has acquired a lock on said message queue;

means for comparing said use count with a previously read use count; and

means responsive to a successful comparison of said use count with said previously read use count and operating atomically with said comparing means for updating said use ~~count;~~ count to indicate a new count of tasks accessing said message queue and ~~means for~~ updating said use count flag ~~atomically with updating said use count~~ to indicate whether said caller has acquired a lock on said message queue.

25. (Cancelled)

26. (Currently amended) The method of ~~claim 25~~ claim 24 in which said use count is stored in a message queue table having an entry for said message queue, said message queue table also storing a pointer to said message queue, said method comprising the further step of:

comparing said pointer with a previously read pointer atomically with said updating operations, said updating operations being performed only if said pointer matches said previously read pointer.

27-33. (Cancelled)

34. (Currently amended) A program storage device readable by a machine, tangibly embodying a program of instructions executable by the machine to perform method steps for performing a recoverable operation on a message queue in response to a request by a caller in an information handling system, said method steps comprising:

storing a use count for said message queue indicating a count of tasks accessing said message queue;

storing a use count flag for said caller indicating whether said caller has acquired a lock on said message queue;

comparing said use count with a previously read use count; and

if said use count matches said previously read use count, then, atomically with said comparing step, updating said use ~~count,~~ count to indicate a new count of tasks accessing said message queue and ~~atomically with updating said use count,~~ updating said use count flag to indicate whether said caller has acquired a lock on said message queue.

35. (Cancelled)

36. (Currently amended) The program storage device of ~~claim 35~~ claim 34 in which said use count is stored in a message queue table having an entry for said message queue, said message queue table also storing a pointer to said message queue, said method steps further comprising:

comparing said pointer with a previously read pointer atomically with said updating steps, said updating steps being performed only if said pointer matches said previously read pointer.

37-43. (Cancelled)